I.Ü.Siyasal Bilgiler Fakültesi Dergisi
No: 25 (Ekim 2001)

YAPAY SINIR AGLARI

Ögr. Gör.Dr. Melda AKIN*

Abstract
In this paper, artificial neural networks which is a new computational paradigm which can be also be considered as a new statistical modeling and estimation tool, is described. Artificial neural networks have been successfully applied to the problems where there is a nonlinear mapping between the inputs and the outputs such as pattern recognition, control and time series modeling, etc. First, the history then the artificial neuron models is explained. Then, the multiplayer perceptron type artificial neural network and the learning algorithm, i.e., error backpropagation algorithm are given.

1. GIRIS

Son zamanlarda özellikle dogrusal olmayan gönderim problemleri ve modellemede yapay sinir aglari (YSA) denetimden ekonometrik modellemeye kadar bir çok alanda basari ile kullanilmaya baslanmistir. Bu makalede arastiricilara bir baslangiç yapabilmeleri için temel bilgiler verilerek yapay sinir aglari tanitilacaktir.

Yapay sinir aglari bir hesaplama yöntemidir ve çikis noktasi beynin davranislarinin modellenmesidir. YSA geleneksel bilgisayar sistemlerinde kullanilandan tamamen farkli bir bilgi isleme teknigidir. YSA'lar ayni zamanda bir çok veri isleyip sonuç üreterek çok zor ve baska yöntemlerle çözülmesi imkansiz problemleri çözebilirler. Genellikle, bilgisayarlarin güvenilirligi ile insan uzmanliginin beklentilerini birlestirmek amaciyla kullanilirlar.

YSA'lar bu güne kadar basta örüntü tanima (pattern recognition) ve denetim kurami ve zaman serisi modelleme olmak üzere çok sayida alanda basarili bir sekilde uygulanmislardir. Bu baglamda YSA'lar bir istatistiksel modelleme ve kestirim yordamlari sinifi olarak da gözönüne alinabilir. Bu tür uygulamalarda amaç, bir egitim veri kümesine bagli olarak bu verileri üreten istatistiksel süreci modellemek ve yeni verilerle en iyi kestirimi yapabilmektir. Istatistiksel modelleme problemleri üç gruba ayrilabilir: yogunluk kestirimi, siniflandirma ve regresyon. Bu nitelikleriyle YSA'lar sosyal bilimlerde de bir çok uygulama alani bulmaktadir.

---

* Istanbul Üniversitesi, Siyasal Bilgiler Fakültesi, Ögretim Görevlisi.

Makalenin ikinci bölümünde YSA'larin kisaca tarihçesi verilmekte, üçüncü bölümde YSA modelleri, dördüncü bölümde baglantilar, besinci bölümde ögrenme yöntemleri, altinci bölümde ise en çok kullanilan YSA tipi olan çok katmanli algaçlar anlatilmaktadir. Yedinci bölümde ise sonuçlar verilmektedir

2. TARIHÇE

Modern yapay sinir aglari dönemi Mc Cullogh ve Pitts'in (1943) öncü çalismasiyla baslamistir. Bu makalede nörofizyoloji ve matematiksel mantik konusundaki çalismalari bir araya getiren yapay sinir agi modeli ve ilgili kuram ortaya konmustur. Bu çalisma hem yapay sinir aglari hem de yapay zeka disiplinlerini doguran çalisma olarak kabul edilir.

YSA alanindaki bir sonraki önemli gelisme 1949'da Hebb'in kitabinin yayinlanmasiyla oldu. Ilk defa bu çalismada sinaptik degisim için fizyolojik ögrenme kurali verilmistir.

'Beyin için Tasarim' isimli kitap 1952'de Ross Ashby tarafindan yayinlanmistir. Kitap canlilardaki sinir sisteminden etkilenerek yazilmistir.

Rosenblatt (1958) algaç (perceptron) adli modeli ve daha sonra bu modelin egitimiyle ilgili kurami gelistirince YSA'lar özellikle örüntü tanima alaninda uygulanir hale geldi ve uzun bir süre de her sorunun çözümünde kullanilabilecegi varsayildi. Minsky ve Papert(1969) bu modelin yeterli olmadigini gösterdiler. Bu da bu konuya ayrilan kaynaklarin hemen tamamen kesilmesine neden oldu, maddi yetersizlikler nedeniyle arastirmalar azaldi.

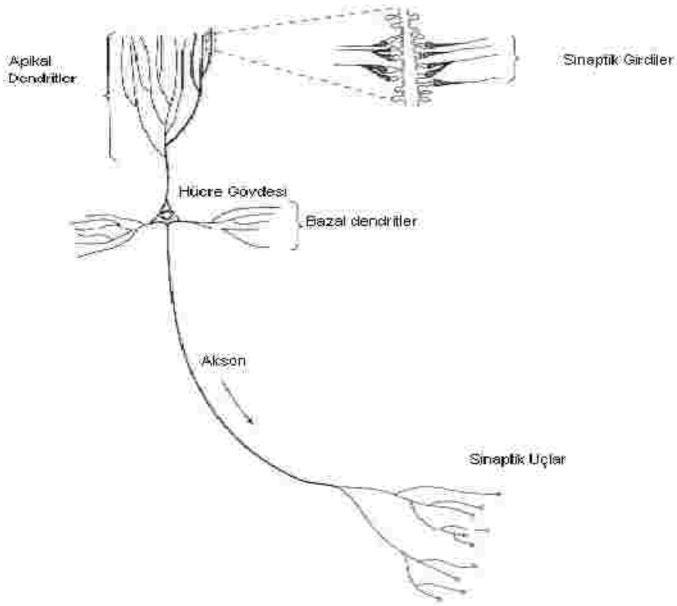
1982'de Hopfield'in bir arastirmasi yeni gelismelere olanak tanimistir. Modelin eksikliklerini gideren çok katmanli algaç(multi-layer perceptron) modeli ve hata geri yayma yordami (error back propagation) farkli disiplinlerden birçok bilim adaminin olusturdugu bir grubun yayinladigi kitapla (Rumelhart ve McCleland, 1986) ortaya konmus ve sürekli gelistirilerek bugün de basariyla kullanilmaktadir.

3. YAPAY SINIR AGLARI MODELLERI

3.1. Nöronlar

YSA'nin ne oldugunu ve ne yapabildigini anlamak için beyindeki nöronlarin (beyin hücreleri) incelenmesi gereklidir.

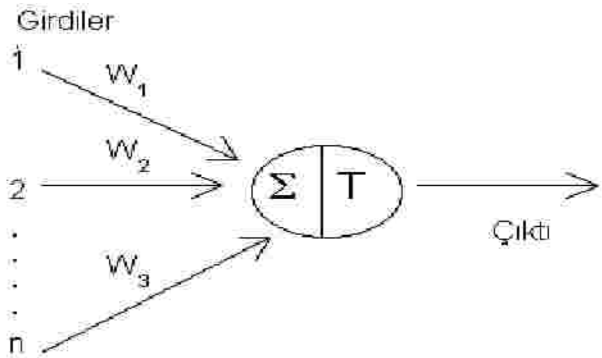
Sekil 1'de insan beynindeki tipik bir nöronun yapisi gösterilmektedir. Beynimiz farkli tipte yaklasik olarak 10^{11} nörondan olusmaktadir. Çalismasi ise basit olarak söyledir. Girdi sinyalleri dendritler yoluyla toplanir, üretilen çikti akson yoluyla sinapslara gönderilir. Çikti, girdi sinyallerinin toplamina baglidir. Toplam belli bir esik degerinden daha büyükse nöron tetiklenir ve elektrik sinyalleri akson yoluyla sinaptik baglantilara gider. Sinapslar diger nöronlarin dendritlerine baglidir. Nöronlar binlerce baglanti yardimi ile birbirlerine bagli olduklari için bir ag yapisi olustururlar.



Sekil 1: Nöron (Haykin 1999)

3.2. Yapay Sinir Agi Islem Birimleri

Sekil 2'den görüldügü gibi YSA islem birimi de dogal nöronlara benzerlik gösterir. Bir islem biriminin girdilerinin sayisi, diger islem birimlerinin sayisina ve bunlarin söz konusu islem birimine olan baglantilarina baglidir. Her girdinin bir agirligi vardir



Sekil 2: Ideal yapay nöron

YSA'daki bir islem birimi söyle çalisir.

Adim 1. Ilk olarak agdaki diger islem birimlerinden gönderilen toplam sinyal hesaplanir.

Adim 2. Içsel etkilenimin düzeyi etkilenim islevi kullanilarak belirlenir.

Adim 3. Birim agdaki diger islem birimlerine sinyal gönderilir, bu sinyal birimin içsel etkileniminin bir islevidir. Bir islemciden digerine sinyal gönderme islemi agirlikli baglantilar üzerinden yapilir.

3.3. Baglantilar

Her bir YSA'da çok sayida islem birimi vardir. Degisik hesaplamalarda farkli ag yapilari kullanilir. Genel olarak baglantilarina göre YSA'lar iki sinifa ayrilir: ileri-beslemeli ve geri-beslemeli ag yapilari . Ileri beslemeli yapida baglar tek yönlü olup çevrimler yoktur. Geri beslemeli bir yapida ise baglar her hangi bir topoloji olusturacak sekilde seçilebilir. Ileri beslemeli ag yapilarinda birimler genellikle katmanlar olusturacak sekilde yerlestirilirler. Böyle bir yapida her birim yalnizca kendinden sonraki katmandaki birimlere baglanir, ayni katmandaki birimlere ya da daha önceki katmana baglanti yoktur.

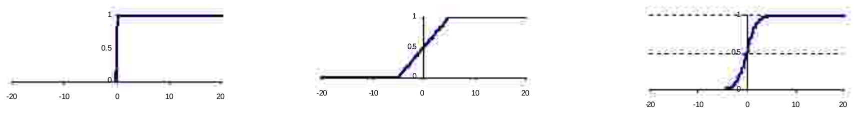
Ag yapisinda çevrimler olmazsa hesaplama düzgün olarak girdilerden çiktilara dogru ilerlenerek yapilabilir. Bu durumda bir önceki zaman adimindaki etkilenim hesaplamada hiçbir rol oynamaz. Çünkü, daha önceki bir birime geri beslenmez. Dolayisiyla ileri beslemeli bir YSA'nin agirliklari disinda bir iç durumu yoktur. Sadece o anki girdilere ve agirliklara bagli olarak çikti hesaplanir.

Geri beslemeli bir yapida ise önceki zaman adimlarindaki verilerde çiktiyi etkiler. Bu yüzden bu tür yapilar kararsiz hale gelebilir ya da kaotik bir davranis gösterebilir. Dolayisiyla da bir veri girildiginde kararli bir çikti elde etmek çok uzun süre alabilir ve ögrenme güçlesebilir. Ancak bellekli islevler gibi daha karmasik islevler ileri beslemeli yapilardan daha çok geri beslemeli yapilar kullanilarak modellenebilir.

3.4. Etkilenim islevleri

Degisik YSA modellerinde degisik etkilenim islevleri kullanilir. Sekil 3'de en çok kullanilan islevler olan basamak, rampa ve sigmoid islevleri gösterilmistir.

Basamak islevinin bir esigi vardir. Esik, birimi tetikleme için gerekli olan en düsük agirlikli girdi toplamini gösterir Girdi esikten büyük oldugu zaman çikti 1 olur, aksi halde ise 0 olur. Biyolojik olarak 1 nöronun tetiklenmesi, 0 ise tetiklenmemesine karsilik gelir. Çogunlukla esik yerine fazladan bir girdi agirligi kullanmak matematiksel islemleri kolaylastirir.



$$f(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \qquad f(x) = \begin{cases} 0 & x < a \\ mx + n & a \leq x \leq b \\ 1 & x > b \end{cases} \qquad f(x) = \frac{1}{(1 + e^{-bx})}$$

(a) (b) (c)

Sekil 3: Etkilenim islevleri. (a) Basamak islevi, (b) Rampa islevi, (c) Sigmoid islevi.

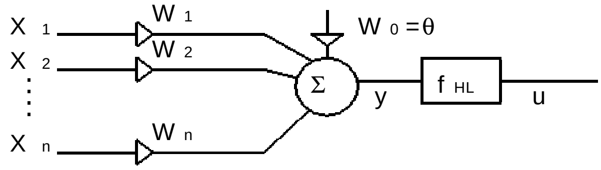
Sigmoid islevi ise süreklidir ve x'nin degeri $-\infty$ to ∞ arasinda degisirken islevin degeri 0'dan 1'e monotonik olarak artar. Sigmoidin kazanci β, geçis bölgesinin dikligini belirler. Kazanç sonsuza giderken sigmoid de esik islevine yaklasir. Sigmoidin avantajlarindan birisi türevinin her noktada tanimli olmasidir. Bu özellik sayesinde çok katmanli yapay sinir aglari için gradyan temelli bir ögrenme yordami gelistirmek mümkün olmustur.

3.5. Yapay Sinir Alarinin Islev Yaklastirma Özelligi

YSA'larin önemi genel amaçli bir dogrusal olmayan girdi-çikti gönderim araci olarak kullanilabilmelerinden kaynaklanmaktadir. Evrensel Yaklasim Teoremi (Hornik, Stinchcombe ve White, 1989) tek katmanli bir ÇKA'nin verilen bir $x_1,...,x_m$ girdi kümesi ile istenen çiktiya $f(x_1,...,x_m)$ düzgün bir ε yaklasim hesaplamaya yeterli olacagini göstermektedir. Ancak teorem tek katmanin ögrenme süresi, uygulama kolayligi ve genellesme açisindan optimum oldugunu söylememektedir. Bu nedenle YSA'larin ag yapisinin (kaç sakli katman olacagi, her katmanda kaç birim olacaginin) belirlenmesi zor bir problem olarak ortaya çikmaktadir.

3.6. Çok Katmanli Algaç Modeli

Bu bölümde en çok kullanilan YSA tipi olan çok katmanli algaç (ÇKA) tanitilacaktir. Algaç modelinde girdi n-boyutlu bir vektördür. Algaç yukarida belirtildigi gibi girdi vektörünün bütün bilesenlerinin agirlikli bir toplamini hesaplar ve bu toplamdan bir esik degerini (q) çikarir. Bu islemin sonucu dogrusal olmayan bir etkilenim islevine girdi olarak beslenir (Sekil 4).



Sekil 4: Algaç

Hesaplamalarda kolaylik olmasi için esik degeri de bir agirlik olarak alinir (w_0=q,). Bu agirlik, degeri her zaman +1 olan bir ek girdinin agirligi olarak kullanilir. Agirlikli toplam bu sekilde genisletilmis girdi vektörüyle agirlik vektörününü iç çarpimi olarak ifade edilebilir. Algaç tarafinda yapilan islem asagida verilmistir:

$$y = w^T . x \qquad (1)$$
$$u = f_A (y)$$

Algaçlar katmanlar halinde arka arkaya yerlestirilince genellikle dogrusal olmayan islev olarak sigmoid (lojistik) islev kullanilir:

Çok katmanli algaçta üç degisik tip katman bulunur:

§ Girdi Katmani. Bu katman agin disindaki kaynaklardan girdileri alan birimlerden olusur. Bu birimlerin özel bir aktarma islevleri yoktur, girdilerini çiktilarina hiç degistirmeden aktarirlar.

§ Sakli Katmanlar. Bir YSA'da bir veya daha çok sakli katman bulunabilir. Bu katmanlar girdi katmaniyla çikti katmani arasinda yer alirlar. Girdi ya da çikti katmani degildirler.

§ Çikti Katmani. Bu katman dis ortama isaret gönderen birimlerden olusur. Bu birimlerin çiktilari ayni zamanda agin çiktilarini da olusturur.

Girdi vektörü birinci katman algaçlarina beslenir, bu katmanin çiktilari bir sonraki katmanin birimlerine girdi olusturur, v.b. Genellikle l. katmanin bütün birimleri l+1. katmandaki bütün birimlere baglidir. Çok katmanli algaçlar sakli katman sayilarina göre siniflandirilirlar.

Son katmanda birden fazla çikti birimi olmasi genellikle örüntü tanima uygulamalarinda raslanan bir durumdur. Bu tür uygulamalarda çikti birimleri siniflara karsilik gelir.

4. ÖGRENME YÖNTEMLERI

YSA'larin mimarilerinin belirlenmesi ve baglanti agirliklarinin hesaplanmasi için ögrenme yöntemleri adi verilen ve yapay zekada örneklerden ögrenme yöntemi olarak tanimlanan tümevarimsal yöntemler kullanilir (Russell ve Norvig 1995). Bu yöntemlerde ögrenmesi beklenen dizgeye ögrenilecek kavramla ilgili örnekler gösterilir ve dizgenin bir genellemeye giderek istenen bir kavrami olusturmasi beklenir. Örneklerden ögrenme üç sekilde yapilabilir.

- Gözetmenli Ögrenme: Bu ögrenme yönteminde verilen bir girdi için istenen çikti degeri bir gözetmen tarafindan dizgeye bildirilir. Degisik örnekler verilerek, dizgenin çiktisiyla istenen çikti arasindaki farki belirli bir ölçüte göre en aza indirmesi beklenir.
- Kuvvetlendirmeli Ögrenme: Bir dizi ardisik girdiye karsilik çikti ürettikten sonra dizgeye basarili basarisiz seklinde nitel bir bilgi verilir. Bu bilgiyi kullanarak dizgenin basarimini yükseltmesi beklenir. Bu yöntem gözetmenli ögrenmenin özel bir durumudur.
- Gözetmensiz Ögrenme: Burada bir gözetmen bulunmayip yapay sinir aginin, girdilerin içindeki iliskileri ve düzenlilikleri bulmasi beklenir.

YSA'larda her üç yöntemde kullanilmakta ise de uygulamalarda en çok gözetmenli ögrenme yöntemi kullanilir. Asagida çok katmanli algaçlar için anlatilacak olan ögrenme yöntemi buna tipik bir örnektir.

4.1. Çok Katmanli Algaçlar Için Ögrenme Yöntemleri

YSA'larda gözetmenli ögrenmede en sik raslanan yaklasim bir gradyan azalimi (gradient descent) yordami kullanilmasidir. Bir ÇKA'nin ögrenme asamasinin klasik kosulsuz, dogrusal olmayan bir optimizasyon problemi olarak ele alinabilinecegi gösterilmistir (Topçuoglu ve Akin 1994). Bu tür bir problemin çözümü tipik olarak bir amaç islevini en büyüklemek ya da en küçüklemek için bir küme bagimsiz degiskeni yinelemeli olarak degistirmek seklindedir. ÇKA bu modele gayet uygundur: agirliklar bagimsiz degiskenler olarak ve egitim kümesi üstündeki hatalarin karelerinin toplami da amaç islevi olarak alinabilir. Hata ölçütü ya da amaç islevi asagidaki gibi tanimlanir:

$$E(w)=\frac{1}{2}\sum_{im}(V_i^m-O_i^m)^2=\frac{1}{2}\sum_{im}(V_i^m-\sum_k w_{ik}e_k^m)^2 \quad (2)$$

Burada O_i^m ÇKA'nin çiktisi, V_i^m her i ve m için istenen çikti. e_k^m m. örüntü için girdiyi göstermektedir.

Agi egitmenin amaci, agirliklari, sonuçta her örüntü için agin çiktisinin istenen çiktiya esit olacagi sekilde degistirmektir. Bir çok optimizasyon yöntemi olmasina karsin her birinin hareket noktalari baskadir (Murray 1979), (Fletcher 1991). Bazi yöntemler yalnizca en küçüklenecek islevin hesaplanmasini gerektirirken digerleri bu islevin bagimsiz degiskenlere göre kismi türevlerinin de hesaplanmasini gerektirmektedir. Çok Katmanli Algaç durumunda gradyan denklem (2)'deki $E(w)$'nin agirliklara göre kismi türevlerinden olusur. Egitimde kullanilabilecek dogrusal olmayan optimizasyon yöntemleri asagidaki gibi siralanabilir:

§ Gradyan Azalmasi (Gradient Descent) yöntemi

§ Eslenik Gradyanlar (Conjugate Gradients) yöntemi

- Newtonumsu (Quasi Newton) yöntemler
- Levenberg-Marquardt yöntemi

Bazi yöntemler bilgisayarla hesaplamada N^2 mertebesinde yer gerektirirken digerleri N mertebesinde yer gerektirir. Burada N boyut (agirlik) sayisidir. N'in küçük degerleri için aradaki fark pratik açidan önemli degilken bazi problemler için gereken bellek miktari çok büyüyebilir.

Genel bir islevin bir asaç(extremum) noktasi yerel ya da global olabilir. Dogrusal sistemlerin disinda herhangi bir dogrusal olmayan problemin global asaç noktasinin bulunmasi çok zor bir problemdir.

Yerel minimumu hesaplamak için sik kullanilan bir yöntem, bir çok rasgele baslangiç noktasindan baslamak ve çok boyutlu uzaydaki bir yüzey boyunca bir ilerleme kuralina göre bir asaç noktasina ulasana kadar ilerlemektir. Bu yöntemlerde, bulunan asaç noktalarinin en uçtakinin seçilmesiyle optimizasyon probleminin çözümünün bulunacagi varsayilir. Bu stratejinin ana fikri, $t+1$ anindaki yeni vektör olan x^{t+1}'in t anindaki vektör olan x^t cinsinden yinelemeli olarak asagidaki sekilde hesaplanabilmesidir:

$$x^{t+1} = x^t - k.B.g \qquad (3)$$

burada g, x^t 'de hesaplanan gradyandir B gradyanin bir dönüsüm matrisi olup bir arama yönü gösterir ve k da adim boyudur. Bütün optimizasyon yöntemleri k ve B'yi seçmek için degisik yöntemler önerir. Gradyanin bilesenleri

$$g_j = \frac{\partial E(x)}{\partial x_j} \qquad (4)$$

seklinde ifade edilebilir.

4.2. Gradyan Azalmasi Yordami

Gradyan azalmasi yordami, rasgele bir baslangiç noktasindan baslayarak her girdi örüntüsü için YSA'nin tam istenen çiktiyi üretmesini saglayacak bir agirlik kümesinin ardisik iyilestirmelerle bulunmasini saglayan bir yöntemdir.

Hata ölçütü $E(w)$ (2) denklemindeki sekilde verilince, bir agirlik kümesi w_{ik}'yi w uzayinda tanimladigi yüzeyden yokus asagiya ilerlenerek iyilestirilebilir. Gradyan azalmasi yordami her w_{ik}'yi E'nin mevcut konumdaki gradyaniyla orantili bir Δw_{ik} miktari kadar degistirme prensibine dayanir:

$$\Delta w_{ik} = -h \frac{\P E}{\P w_{ik}} \qquad (5)$$
$$= h \sum_{m} (V_i^m - O_i^m) e_k^m$$

Bu denklemi (3) denklemiyle karsilastirirsak burada k=h ve B=1 oldugu görülür. Eger agirliklarda (5) denklemiyle verilen degisiklikleri her bir girdi örüntüsü e_k^m için sirayla yaparsak m. örüntü için gerekli degisiklik asagidaki gibi hesaplanabilir:

$$\Delta w_{ik} = h(V_i^m - O_i^m) e_k^m \qquad (6)$$

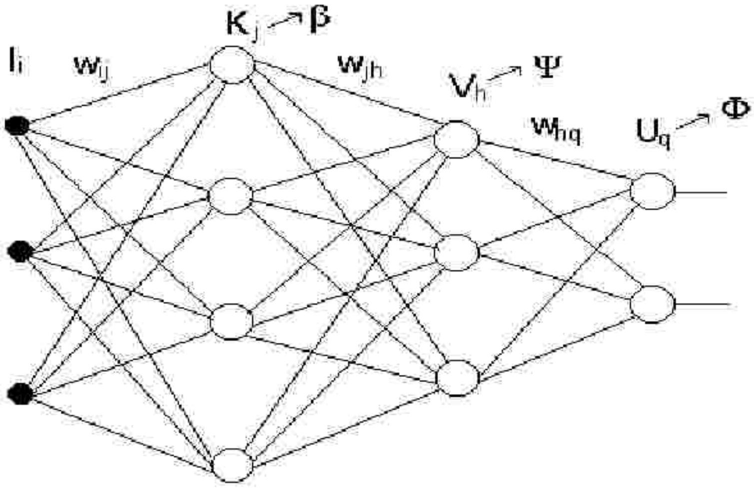
Eger hatalari (ya da deltalari) $d_i^m = V_i^m - O_i^m$ seklinde tanimlarsak,

$$\Delta w_{ik} = h d_i^m e_k^m \qquad (7)$$

elde edilir. Bu denklemlerdeki sonuca genellikle Delta Kurali, kurali adi verilir.

4.3. Hata Geri Yayma Yordami

Hata geri yayma yordami gradyan aramaya dayanan ve bir amaç islevini enküçükleyecek ag agirliklarini bulmaya yarayan ve programlamasi ve anlasilmasi daha kolay oldugu için en sik kullanilan yordamdir. Sekil 5'de denklemlerde kullanilacak semboller gösterilmektedir.



Sekil 5: Çok katmanli algaç.

Amaç islevini veren denklem (2) asagidaki sekilde yeniden yazilabilir:

$$E(w) = \sum_{P=1}^{N_p} E_P(w) \tag{8}$$

Burada N_p egitim örüntüsü sayisini, $E_p(w)$ ise p. örüntü için hatalarin karesinin toplamini göstermektedir. $E_p(w)$ asagidaki sekilde yazilabilir:

$$E_P(w) = \frac{1}{2}\sum_{q=1}^{N_q} (d_q^p - \Phi_q^p)^2 \tag{9}$$

Burada N_q çikti katmanindaki birim sayisini, d_q^p q. birimin p. örüntü için istenen çiktisini, Φ_q^p ise q. birimin p. örüntü için gerçek çiktisini gçstermektedir. Yukaridaki iki denklem birlestirilerek

$$E(w) = \frac{1}{2}\sum_{p=1}^{N_p}\sum_{q=1}^{N_q} (d_q^p - \Phi_q^p)^2 \tag{10}$$

yazilabilir. Burada ,

$$\Phi_q^p = \Phi_q^p(u_q^p) \tag{11}$$

çikti islevi olup yukarida belirtildigi gibi genellikle sigmoid olarak kullanilir.

Yordamda kullanilacak bütün kismi türevler zincir kurali kullanilarak hesaplanabilir. Hata geri yayma yordaminda agin agirliklari yinelemeli olarak asagidaki sekilde belirlenir:

$$w_{mn}(t+1) = w_{mn}(t) - \eta \left.\frac{\partial E(w)}{\partial w_{mn}}\right|_{w(t)} \tag{12}$$

$$w_{mn}(t+1) = w_{mn}(t) - \eta \sum_{p=1}^{N_p} \left.\frac{\partial E_p(w)}{\partial w_{mn}}\right|_{w(t)}$$

Burada η ögrenme hizi adi verilen pozitif bir parametredir. E_p'nin agdaki bütün agirliklara göre kismi türevlerini yerine koyarsak her agirlikta yapilmasi gereken degisiklik için asagidaki ifadeyi buluruz:

$$\Delta w_{pq} = \eta \sum_{\text{örüntü}} \delta_{\text{çikti}} V_{\text{girdi}} \tag{13}$$

Burada girdi ve çikti ilgili baglantinin iki uç noktasi p ve q'ye aittir ve V ilgili birimin girdisindeki etkilenimi göstermektedir. δ'nin anlami ilgili katmana baglidir. Burada katsayilar ileri yöndeki agirliklar olan w_{ij} 'lerdir, ancak bu durumda isareti ileri dogru yaymak yerine hatayi geriye dogru yaymaktadirlar. Bu yüzden bu yönteme hata geri yayma yöntemi adi verilir.

Bir ÇKA'yi hata geri yayma yordami kullanarak egitme iki sekilde yapilabilir: yerel güncelleme yöntemi ve birikimli güncelleme yöntemi.

§ Yerel Güncelleme Yöntemi. Bu yöntemde her örüntü m, girdiden aga beslenince hatalar hemen hesaplanir ve agirliklar degistirilir. Bu sekilde her adimda hata islevi azalir (yeterince küçük η için) ve ardisik adimlarin yerel gradyana uymasi saglanir.

§ Birikimli Güncelleme Yöntemi. Bu yöntemde ise agirliklar bütün örüntüler aga beslendikten sonra güncellenir.

Sekil 6'da Hata Geri Yayma Yordami verilmektedir. Bu yordamda agdaki her agirlik için ögrenme parametresi ayni alinmistir. Bazi arastirmacilar ögrenme parametrelerini her katman için farkli hatta her agirlik için farkli seçmektedirler. Genellikle en iyi ögrenme parametresini belirlemek zordur ve deneme yanilma yöntemi kullanilir.

Adim 1. Agirliklara küçük rassal ilk degerler ata..

Adim 2. Bir örüntü seç ve girdi katmanina besle.

Adim 3. Isareti ag boyunca asagidaki denklemi kullanarak her i ve m için agin çiktilari hesaplanincaya kadar ileriye yay.

$$V_i^m = g\left(\sum_j w_{ij}^m V_j^{m-1}\right) \tag{14}$$

Burada, V_i^m m. katmandaki i.birimin çiktisini w_{ij}^m ise V_j^{m-1} ile V_i^m arasindaki baglantinin agirligini göstermektedir. m hesaplamanin yapildigi katmandir.

Adim 4. Çikti katmaninin deltalarini hesapla.

Adim 5. Önceki katmanlarin deltalarini hatalari geri yayarak hesapla.

Adim 6. Denklem (13)'ü kullanarak bütün baglantilari asagidaki sekilde güncelle.

$$w_{ij}^{yeni} = w_{ij}^{eski} + \Delta w_{ij} \tag{15}$$

Adim 7. Adim 2'ye git ve bir sonraki örüntü için islemi tekrarla.

Sekil 6: Hata geri yayma yordami.

Gradyan azalmasi, η küçükse ögrenme islemi çok yavas ilerler. η'nin çok büyük oldugu durumlarda da agirliklarin degerlerinde büyük salinimlar görülür. Bu sorunun üstesinden gelmek için kullanilan yöntemlerden biri bir momentum terimi eklemektir.

Buradaki düsünce her baglantiya bir eylemsizlik ya da momentum vermektir. Bu sekilde agirliklarin degerlerinin her güncellemede büyük salinimlar yapmasi yerine ortalama yokus asagi "kuvvet" dogrultusunda degisme egilimi göstermesi beklenir.

Bu düzeltme her agirlik degisimine bir önceki adimdaki degisiklikten bir katki ekleyerek yapilabilir:

$$\Delta w_{pq}(t+1) = -\eta \frac{\partial E_p}{\partial w_{pq}} + \alpha \Delta w_{pq}(t) \qquad (16)$$

Momentum parametresi (α) 0 ile 1 arasinda olmalidir. Uygulamada genellikle 0.9 degeri seçilir.

Yordamin durmasi için kullanilabilecek bir kaç ölçüt vardir.

- Gradyanin bir minimum degerin altina düsmesi. Burada gradyanin büyüklügünü temel olarak alinir. Yordam gradyanin büyüklügü yeterince küçükse durdurulabilir, çünkü tanim geregi minimumda gradyan sifir olacaktir.
- Hatanin bir minimum degerin altina düsmesi. Ikinci olarak $E(w)$'nin belli bir degerin altina düsmesi halinde yordam durdurulabilir.
- Yineleme sayisi limitine ulasilmasi. Üçüncü olarak da belli sayida yineleme yapilinca isleme son verilebilir. Bu durumda yordamin minimuma ulasma garantisi yoktur.
- Çapraz Geçerleme. Dördüncü yöntem ise çapraz geçerleme (cross validation) yöntemidir.Veri, agin egitiminde kullanilan bir egitim kümesi ve agin genelleme basarimini ölçmede kullanilan bir deneme kümesi olarak iki kümeye ayrilir. Ögrenme sirasinda agin egitim kümesindeki basarimi sürekli artmaya devam etmesine karsin deney kümesindeki basarimi bir noktaya kadar arttiktan sonra azalmaya baslar. Bu noktada ag egitim kümesini ezberlemeye baslamis demektir ve egitim kümesinde bulunmayan örüntüler için genelleme yetenegini yitirir. Bu yüzden egitim bu noktada durdurulmalidir. Ilk üç ölçüt parametrelerin seçimine çok hassastir ve iyi bir seçim yapilmazsa sonuçlar çok kötü olabilir. Çapraz geçerleme bu seçimden etkilenmez.

5. SONUÇ

Bu makalede çesitli istatistiksel uygulamalarda kullanilabilecek yeni bir hesaplama yöntemi olan yapay sinir aglari tanitilmistir. Özellikle borsalarda zaman serilerinin modellenmesinde ve genel olarak yeterli veri bulunan her çesit dogrusal olmayan regresyon uygulamalarinda çok basarili sonuçlar veren YSA'larin sosyal bilimler alaninda da kullaniminin hizla artmasi beklenmektedir.

KAYNAKLAR

- Ashby, W. R., 1952. Design for a Brain, New York:Wiley.
- Fletcher R., 1991. Practical Methods of Optimisation, John-Wiley & Sons, London.
- Haykin, S. 1999. Neural Networks: A Comprehensive Foundation, Ikinci Baski, Upper Saddle River, NJ: Prentice-Hall.
- Hebb, D.O., 1949. The Organization of Behavior:A Neurophysiological Theory, New York:Wiley.
- Hopfield, J.J, 1982. "Neural networks and physical systems with emergent collective computational abilities," Proceedings of the National Academy of Sciences, USA, vol. 79, pp.2554-2558.
- Hornik K., M. Stinchcombe ve H. White, 1989. "Multi-Layer Feedforward Networks Are Universal Approximators," Neural Networks, Cilt. 2, pp.35-366.
- McCullogh, W.S. ve W. Pitts, 1943. "A logical calculus of the ideas immanent in nervous activity," Bulletin of the Mathematical Biophysics, vol. 5, pp.115-133.
- Minsky, M.L. ve S. A. Papert, 1969. Perceptrons, Cambridge, MA: MIT Press.
- Murray, W., 1979. Numerical Methods for Unconstrained Optimisation, Academic Press, London.
- Rosenblatt, F., 1958. "The Perceptron: A probabilistic model for information storage and organization in the brain," Psychological Review, vol. 65, pp.386-408.
- Rumelhart, D.E. ve J.L McCleland, ed., 1986. Parallel and Distributed Computing: Explorations in the Microstructure of Cognition, vol.1-2, Cambridge:MA:MIT Press.
- Russell, S.J., ve P. Norvig, 1995. Artificial Intelligence: A Modern Approach, Upper Saddle River, NJ: Prentice-Hall.
- Topçuoglu, H. ve H. L. Akin, 1994., "Comparison of Learning Algorithms," 3. Yapay Zeka ve Yapay Sinir Aglari Sempozyumu.